Buon Natale e Felice Anno Nuovo!

IQ3TS on air..!

Microcontrollori per Hobbisti
Cena conviviale di fine mandato
Addestramento di Prot. Civile ARI 2012
OPERA, nuova modalità digitale di QRSS

A.R.I. Sezione di Trieste
Via Pasteur, 16/1
34139 TRIESTE (TS)

Telefono: 0409896119
Fax: 0409890588
e-mail: iq3tsonair@aritrieste.it
web: www.aritrieste.it

Associazione Radioamatori Italiani - TRIESTE NÁ10 DICEMBRE 2012

Il Direttivo della Sez. A.R.I. di Trieste:
Presidente IV3KAS
Vicepresidente IV3TRK
Segretario IV3OTE
Tesoriere IV3NDR
Consiglieri IV3TPW - IV3XHA
Collegio dei sindaci
IV3BKO - IV3WSJ - IV3YAO

IQ3TS on air non costituisce una testata giornalistica, non ha, comunque, carattere periodico e viene pubblicato secondo la disponibilit¨ e la reperibilit¨ dei materiali. Pertanto, non pu¸ essere considerato in alcun modo un prodotto editoriale ai sensi della L. n. 62 del 7.03.2001.

Tutti i progetti vengono pubblicati per libera volont¨ dei singoli autori e non a scopo di lucro. I componenti del Direttivo della Sezione ARI di Trieste e gli autori degli articoli, non si assumono nessuna responsabilit¨ esplicita o implicita riguardante qualsiasi evento o situazione possa verificarsi nel realizzare ed utilizzare gli schemi e le pubblicazioni riportate in queste pagine.

Editoriale

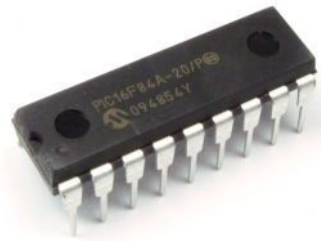
La Redazione, Mauro, IV3WSJ.

Impaginazione del giornalino a cura di Mauro Cok - IV3WSJ
Copertina e banner a cura di Luigi Popovi¸ - IV3KAS

MICROCONTROLLORI per HOBBISTI

di Matteo Copetti - IV3YNB

Salve a tutti; io non sono un programmatore, non professionista per lo meno, sono inoltre abbastanza sicuro che molti lettori non sono troppo "dentro" l'argomento, bench® curiosi. Di trattati sull'argomento ce ne sono moltissimi disponibili in rete, in libreria ed in edicola, vorrei qui tentare di non fare un copia/incolla, perci¸ descriver¸ la mia esperienza con i microcontrollori in maniera non troppo tecnica, sperando di suscitare quantomeno interesse. Rimando alla rete la spiegazione approfondita di cosa siano i microcontrollori: una capatina su Wikipedia ed il gioco ¯ fatto. Vi basti sapere che i microcontrollori (altrimenti detti MCU: Micro Controller Unit) possono essere ridotti al concetto di piccoli computer fatti per eseguire un numero limitato di operazioni e, di solito, sempre quelle. Operazioni comunque non di poco conto, che sfruttano caratteristiche interne all'MCU come convertitori A/D, timers, eeprom, ecc... che troviamo oramai implementati in moltissimi elettrodomestici di casa e non solo. In un momento di noia, vista la mancanza ormai cronica di videogiochi nel mio PC, anni fa me ne andai in giro per il web in cerca di curiosit¨ e chiss¨ cos'altro. Erano i primissimi anni 2000 e non so come mai, approdai in un sito che si fregiava di un titolo altisonante: "PIC by EXAMPLE"!
Il sito si prefissava l'insegnamento della programmazione tramite un classico dei PIC: il 16F84A, e tramite alcuni esempi di linguaggio assembler (spesso indicato come ASM) che venivano proposti in una guida gratuita che scaricai subito. Precisiamo sin d'ora che i PIC sono soltanto una delle tante tipologie di MCU esistenti al mondo e che sono prodotti da una delle tante case produttrici di componenti elettronici. Sono cos³ largamente diffusi e conosciuti che possiamo parlare di PIC per comodit¨, consapevoli che stiamo in realt¨ parlando di MCU. Al tempo della mia scoperta, i PIC erano molto in voga per ñcraccareò i codici della PayTV, quando mi recai da un noto negozio di componenti locale per acquistarne una manciata, il mio amico al banco tent¸ subito di indovinare (sornione) per cosa mi servisse cotanta merce... dovette ricredersi perch¯ io volevo solo far brillare il mio primo led!
Acquistai il mio primo programmatore casalingo da un collega, ed iniziai a scrivere il codice del primo esempio: accensione e spegnimento led. L'ultima mia fatica ¯ invece del 2012: ed ¯ il firmware della logica di comando di un ricevitore supereterodina che, sebbene funzionante, ¯ ancora in fase di sviluppo; perch¯ non c'¯ limite al miglioramento.



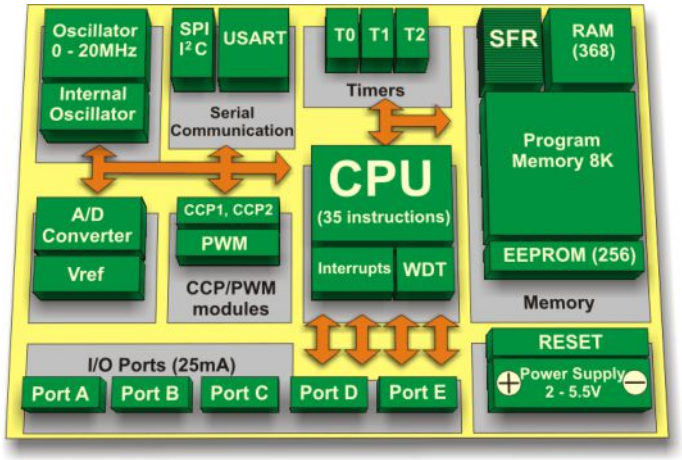
PIC16F84A, la ñnave scuolaò per moltissimi programmatori di MCU

di Matteo Copetti - IV3YNB

Per arrivare a questo punto sono passato per diverse fasi che ritengo siano state molto utili se non fondamentali. Prima fra tutti la dispensa PIC by Example citata prima. Questo PDF, che raccomando a tutti i neofiti, mi introdusse nel mondo del linguaggio ASM. Dal codice pi½ semplice per l'accensione di un LED collegato ad un piedino dell'MCU, alla gestione di un LCD e l'uso degli interrupt nel codice sorgente, questa guida ¯ la sola che serve, almeno all'inizio. Una volta digerita bene e provati tutti gli esempi ivi contenuti si ¯ pronti a fare il passo successivo.
Passo che pu¸ essere inteso come utilizzare un MCU pi½ performante, avente pi½ connessioni e pi½ periferiche onboard (timers, convertitori, porte di comunicazione, maggiore eeprom). Oppure scegliere di programmare qualcosa di complesso che spesso implica la necessit¨ di pi½ connessioni e capacit¨ di contenere codice informaticamente parlando pi½ pesante, perci¸ di un MCU superiore.
Scelsi di utilizzare un PIC16F876 per programmare in ASM un frequenzimetro. Questo mi impose di andare "oltre" la dispensa citata ed iniziare a studiare un sacco di APPLICATION NOTES presenti sul sito della Microchip. Soprattutto per quanto riguardava il soddisfare il flusso dei dati e le operazioni matematiche da eseguire per far girare il frequenzimetro che aveva un contatore a 16bit da trattare attraverso un'architettura ad 8bit, natia del PIC scelto (cosa fattibile ma complessa). Oltre alle application notes, scaricabili dal sito Microchip, anche tramite le applicazioni peer to peer si trova un sacco di materiale gratuito su cui farsi le ossa (datasheet, manuali e dispense gratuite della Microchip, libri, esempi di codice, estratti da forum ecc...).
Si pu¸ di sicuro affermare che bisogna leggere parecchio e capire come raggiungere il risultato prima di scrivere linee di codice. Linee che diventano davvero tante: centinaia e migliaia per operazioni pi½ o

PicStart Plus, programmatore seriale by Microchip

meno complesse. Mentre all'inizio si pu¸ scrivere il codice in notepad, per realizzare progetti importanti si dovr¨ per forza usare un Integrated Development Environment o IDE (Ambiente di sviluppo integrato). Viene spontaneo, per i PIC in ASM usare l'MPLAB della Microchip, ricco di tools di sviluppo, appunto integrati per lo sviluppo, il controllo, il debug e la programmazione dei PIC, magari correlato da un bel programmatore by Microchip che si integri anch'esso con la ñsuite' MPLAB: io ne acquistai uno di seconda mano: il PicStart Plus, che ancora oggi custodisco gelosamente.



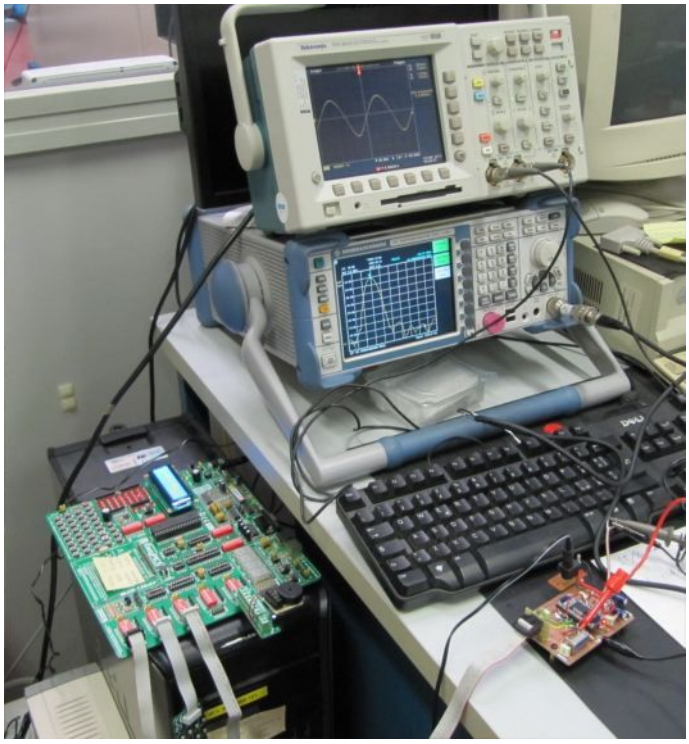
Schema della struttura interna del PIC16F887, MCU con architettura ad 8bit

Si pu¸ per¸ cercare una via un po' pi½ comoda dello scrivere in puro ASM, che ¯ un linguaggio a basso livello, ovvero orientato "alla macchina" piuttosto che "all'uomo". Possiamo scegliere, di scrivere, ad esempio, in basic. Il basic ¯ di sicuro un linguaggio ad "alto livello", pi½ vicino alla sintassi dell'uomo che della macchina. Ci¸ facilita non poco la scrittura del codice necessario allo svolgimento dei compiti dell'MCU, tuttavia il compilatore basic non tratter¨ il codice cos³ bene come l'ASM e si arrogher¨ il diritto di ñgonfiare' il peso del codice finale, perci¸ sappiamo da subito a cosa andiamo incontro. Per il basic bisogna affidarsi ad un nuovo IDE, ce ne sono alcuni in giro, io scelsi mikrobasik della Mikroelektronika, visto che da varie versioni ¯ particolarmente dotato di "supporto cliente" (un forum ricco di persone attive) e presenta un compilatore gratuito fino ad una certa grandezza di file generato, oltre ai soliti tools di sviluppo integrati nell'IDE. Come tradizione vuole, presto superai il limite del demo ed acquistai il compilatore col quale programmo ormai da anni. Di linguaggi ad alto livello ne esistono, ovviamente anche altri, come il C ed il Pascal, supportati da varie marche sviluppatrici di IDE.

MICROCONTROLLORI per HOBBISTI

di Matteo Copetti - IV3YNB

Finora abbiamo parlato di software, per quanto riguarda invece la mia esperienza con l'hardware: ogni qual volta progettavo qualcosa con un PIC, dovevo pur provare il mio programma e vedere come si comportava nel mondo reale. Costruivo delle schede di prototipazione su millefori, una per ogni progetto. Poi scoprii le schede di prototipazione, quelle "vere" con tutti i collegamenti pronti all'uso: socket per ogni tipo di PIC, tutti i pulsanti, tutti i led, per tutte le porte di comunicazione, resistenze di pull down o pull up, magari selezionabili, connessioni esterne, porte usb, seriali, potenziometri per conversioni A/D, LCD, display a 7 segmenti e soprattutto il programmatore onboard. Siccome usavo il software mikroelektronika, acquistai una scheda della stessa marca. Fu uno degli acquisti migliori mai fatti: la compatibilit¨ e l'immediatezza del sistema mi permisero di fare salti da gigante nella programmazione, almeno io vedevo cos³ i miei risultati. Ovviamente il tutto non senza sforzo: c'¯ da sudare anche in basic.



La mia scheda EasyPIC5 e la mia scheda DDS in prova con strumentazione di prima classe

Alla fine possiamo riassumere la mia carriera di programmatore di microcontrollori pic:
prima dispensa + assembler in notepad + programmatore casalingo
mole immensa di dispense e libri + assembler in MPLAB + programmatore PicStartPLUS
mole immensa di dispense e libri + basic + programmatore PicStartPLUS
mole immensa di dispense e libri + basic + scheda prototipazione

Come si capisce dalla lista sintetica, ci¸ che resta sempre fisso sono lôobbligatoriet¨ allôinformazione e alla cultura necessarie a mettere in pratica la teoria: lo studio della materia ¯ di fondamentale importanza. Perci¸ prima di mettere mano al codice, bisogna farsi una bella cultura.
Menzione speciale va assegnata al re delle pubblicazioni: il datasheet dell'MCU. E' la lettura veramente necessaria che non pu¸ mancare nel curriculum del programmatore, spesso le soluzioni le troviamo proprio fra le righe del datasheet. E poi, immancabile, ci deve essere l'idea. E' quella che ci fa iniziare. Che ci fa stare seduti ore a programmare di fronte ad uno schermo di PC, che ci fa intestardire a correggere una porzione di codice affinch® l'MCU operi come vogliamo. Senza le idee, non andremmo da nessuna parte, a volte ne abbiamo cos³ tante da non sapere quale iniziare a sviluppare per prima.

EasyPIC7, lôultima versione delle schede di prototipazione per PIC by Mikroelektronika

Per chi lo desiderasse, il mio indirizzo email ¯: iv3ynb@yahoo.it scrivetemi pure per approfondimenti e/o critiche: qualora vi siano strafalcioni nel mio scritto, vi prego di segnalarmeli senza indugio.

Cena conviviale di fine mandato

di Mauro Cok - IV3WSJ

Gioved³ 14 Novembre, in localit¨ Prosecco, presso la ñTrattoria Socialeò il Consiglio Direttivo e il Collegio dei Sindaci si sono dati appuntamento per una cena conviviale di fine mandato. Lôottima cena e il servizio eccellente del ristoratore, hanno fatto da cornice al clima di serenit¨ e allegria che si respirava tra tutti i presenti. Inevitabili le

battute scherzose che ci hanno accompagnato durante tutta la serata. Questo incontro conviviale, ¯ servito anche a ripercorrere con la mente quello che ¯ stato fatto in questi tre anni di mandato. Tutti si sono adoperati per tener fede agli impegni associativi, in primis il nostro Presidente Luigi, IV3KAS. Tenere unita una squadra di collaboratori credo non sia stato facile, dico credo perch® per me ¯ stata la pri-

ma esperienza come componente del Collegio dei Sindaci, in compagnia di Roberto IV3YAO e con Giorgio IV3BKO Presidente. Personalmente ho avuto piacere di sentirmi parte integrante di questo team e di aver vissuto momenti importanti, perch® propositivi e fruttuosi. Per concludere in bellezza la serata, a tutti i presenti ¯ stato offerto un simpatico presente,

ideato e realizzato dal Presidente Luigi, IV3KAS. A lui, al Direttivo e al Collegio dei Sindaci, un grazie di cuore per questa bellissima esperienzaé !!!

Un augurio va anche al prossimo direttivo, nella speranza che continui e migliori quello che con tanta fatica ed entusiasmo ¯ stato fatto fino ad ora.

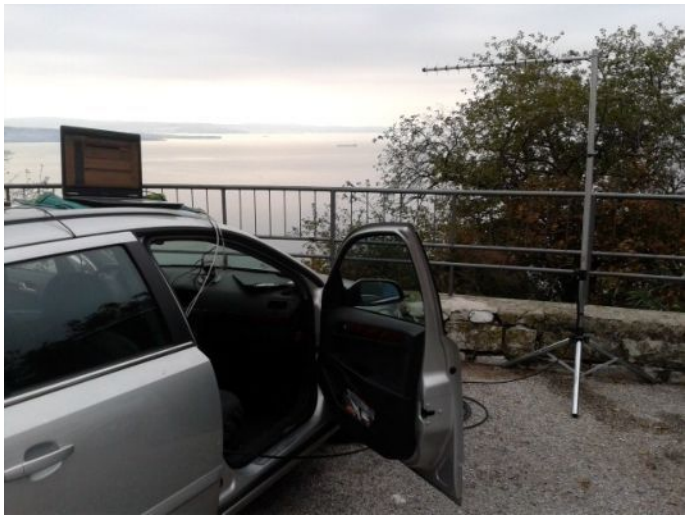
Buon Natale e felice 2013 a tuttié .!!!!!

Mauro, IV3WSJ

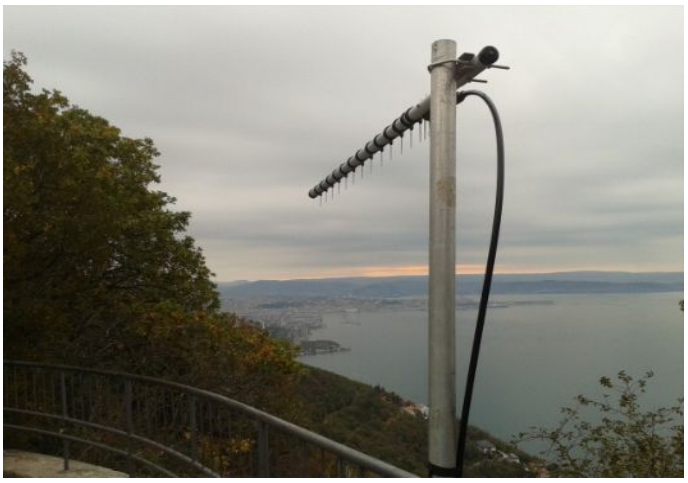
Addestramento di Protezione Civile
ARI 2012

di Roberto Giani IV3TPW

Sabato, 10 Novembre, si è svolto un importante addestramento di Protezione Civile, denominato ARI 2012. Hanno partecipato a questo addestramento moltissimi Radioamatori della Regione, tra questi, molti soci della sezione ARI di Trieste.
Mauro IV3WSJ ed io Roberto IV3TPW, avevamo il compito di inviare dei messaggi con allegate alcune foto in bassa risoluzione, trasmettendo il segnale verso la Prefettura di Trieste con l'apparato della ICOM, l'ID-1, funzionante sulla gamma dei 23cm, 1300Mhz. Arrivati in postazione in località Prosecco Belvedere, dopo aver fatto l'installazione del programma sul PC Portatile, il "difficile" è stato configurare la rete D-Star. Bisogna pensare che è come usare una semplice rete Wi-Fi, anche se la frequenza è diversa e il modem è una vera radio veicolare. Il segnale viene inviato alla Prefettura che a sua volta lo rispedisce al sistema Itanet. Le e-mail vengono gestite direttamente dal WebMail, anche un programma di posta elettronica gestisce regolarmente l'invio e la ricezione di e-mail, le quali possono contenere Testi ma anche Foto a bassa risoluzione.

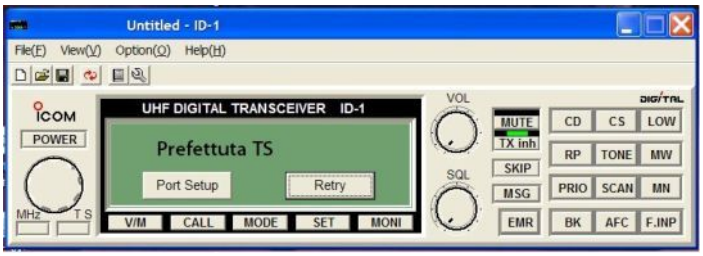


La postazione con tutto l'occorrente preparato da Roberto IV3TPW



La yagi dei 1300Mhz puntata verso la Prefettura

Noi abbiamo inviato due foto per prova a Palmanova e da Palmanova ci hanno inviato una foto dell'interno della stazione. Molto interessante questo sistema di invio dati e immagini che potrebbe risultare utile in caso di emergenza, anche se mi auguro é ..non succeda mai.



Il software per pilotare la radio

73 a tutti é .IV3TPW

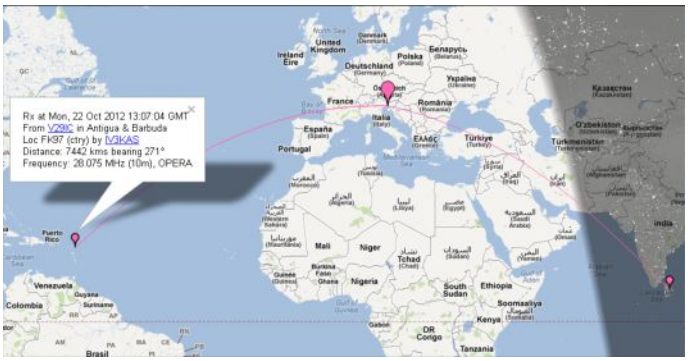
OPERA, una nuova modalit¨ digitale di QRSS

di Luigi Popovi¸ IV3KAS

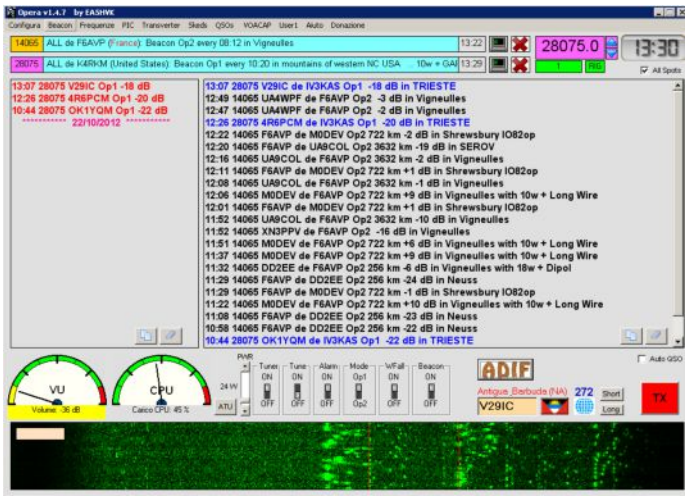
Da alcuni anni mi dedico alle trasmissioni in digitale e recentemente, ho impiegato alcune delle mie ore di tempo libero alla ricerca di nuovi modi di trasmissione. Navigando in internet, sono capitato nel sito di EA5HVK Jos¯, lo stesso che qualche anno faôha creato il software ROS e recentemente, (Gennaio 2012) con lôaiuto di G0NDB ha realizzato il software OPERA. Il software in realt¨ non ¯ altro che un beacon, pi½o meno, come i soliti che siamo abituati ad ascoltare nelle nostre bande. OPERA non ¯ un nuovo modo di fare QSO, almeno per il momento, ma potr¨ esserlo tra breve, questo ¯ quello che dichiara Jos¯. Eôun nuovo modo di usare un beacon personale e anche per vedere lo stato del-

la propagazione, pertanto, si pu¸ trasmette in modo manuale e/o in modo automatico, regolando i propri tempi di trasmissione. Nel caso in cui qualcuno ascolti il nostro segnale, ricever¨ il proprio rapporto tramite il link che si trova allôinterno del software, oppure tramite PSK Reporter. Naturalmente, l'utilit¨ del software, dipende anche dallôesistenza di altre stazioni che eseguono il software automaticamente per poi riferire ci¸ che hanno ricevuto al "reverse beacon", in pratica, quanto succede per lôJT65-HF (vedi) sito web report o CW alla rete Beacon Reverse. Ad ogni modo, per queste funzioni ¯ necessaria una connessione Internet.

Pertanto chi ¯ interessato scarichi dalla rete il software Opera, lôultima versione 1.4.7 che si trova al link: http://rosmodem.wordpress.com/ . Scompattate il file e mettetelo in una cartella, poi lanciate lôeseguibile Opera V. x.y.z.exe (without icons). A questo punto andate al menu Configura/Lingua e scegliete ñitalianoò, proseguite con Configura/operatore compilando tutte le parti; il vostro nominativo e locatore, quindi date la conferma (invio). Continuate con Configurare/radio, cercate nella lista la radio e il vostro modello, e spuntate la parte interessata. Opera ha una caratteristica molto utile, che rileva automaticamente le porte di comunicazione se si sceglie l'opzione CAT. Se invece si desidera utilizzare un semplice circuito PTT piuttosto che CAT, selezionare una porta di comunicazione che sul PC e sulla radio normalmente si usa per le operazioni digitali.



Non mi dilungo oltre nel descrivervi il software, lôinterfaccia ¯ molto intuibile e quindi facile nei principali settaggi, ed essendo in italiano, ¯ molto pi½ semplice. Per chi non conosce lôinglese e desidera far funzionare il tutto nel modo pi½ regolare, non ci sono particolari difficolt¨. Unôaltra info importante, nel caso usiate eQSL e/o un log on-line, inserite i vostri account, il sistema provveder¨ automaticamente a inviare la qsl via eQSL e contemporaneamente sar¨ aggiornato anche il vostro log on-line. Le frequenze pi½ usate fino a qualche mese fa erano i 500,0 136,0 e i 10135,0 kHz. Attualmente si possono trovare sta-



zioni beacon anche in altre frequenze dello spettro delle HF. Le Frequenze usate da beacon Opera: 136.0 477,0 500,0 1836,0 3575,0 5289,0 7038,0 10035,0 14065,0 18105,0 21074,0 24925,0 28075,0 50700,0 70093,0 kHz e in USB.

Il mio set-up: Yaesu 2000D 24W, ant. Invt V per i 40m con valore 3 di onde stazionarie alla frequenza di 28075,0 (vedi le foto), pertanto dei 24W ne rimanevano ben pochi!

Buon divertimento a tutti
IV3KAS